WHAT IS CLAIMED IS:

1. A method, comprising:

predicting an execution path of a first conditional branch operation stored in an entry of a trace cache;

in response to predicting said execution path, if a first operation stored in said entry of said trace cache is not in said execution path according to said prediction, assigning to said first operation a non-executable status indicative that said first operation is not in said execution path;

detecting that said prediction is incorrect subsequent to assigning said non-executable status to said first operation;

assigning an executable status to said first operation in response to said detecting, wherein said executable status is indicative that said first operation is in said execution path.

2. The method as recited in claim 1, further comprising preventing said first operation from executing in response to assigning said non-executable status to said first operation.

3. The method as recited in claim 1, further comprising issuing said first operation from a scheduler for execution without refetching said first operation from said trace cache in response to assigning said executable status to said first operation.

4. The method as recited in claim 1, further comprising:

determining a destination of said first operation in response to assigning said executable status to said first operation;

determining that a second operation is dependent upon the destination of said first operation in response to determining said destination; and

configuring said second operation to receive a result from said first operation in response to determining that said second operation is dependent upon the destination of said first operation.

5. The method as recited in claim 4, further comprising storing in a destination list a respective destination specified by each unretired operation, wherein determining said destination of said first operation further comprises accessing said destination stored in said destination list.

6. The method as recited in claim 1, further comprising:

in response to predicting said execution path, if said first operation stored in said entry of said trace cache is in said execution path according to said prediction, assigning said executable status to said first operation;

detecting that said prediction is incorrect subsequent to assigning said executable status to said first operation; and

assigning said non-executable status to said first operation in response to said detecting.

7. The method as recited in claim 6, further comprising:

determining a destination of said first operation in response to assigning said non-executable status to said first operation responsive to detecting that said prediction is incorrect;

determining that a second operation is dependent upon the destination of said first operation in response to determining said destination; and

configuring said second operation to receive a result from a source other than said first operation in response to determining that said second operation is dependent upon the destination of said first operation.

8. The method as recited in claim 1, further comprising predicting an execution path of a second conditional branch operation stored in said entry of said trace cache, wherein said first operation is dependent upon said first conditional branch operation and said second conditional branch operation, and wherein assigning an executable status to said first operation in response to said detecting that said prediction of said first conditional branch is incorrect is dependent upon said first operation being in the predicted execution path of said second conditional branch operation.

9. A microprocessor, comprising:

a trace cache comprising a plurality of entries, wherein each entry is configured to store one or more operations;

branch prediction logic configured to predict an execution path of a first conditional branch operation stored in an entry of a trace cache; and

dispatch logic coupled to said branch prediction logic and to said trace cache and configured to:

if a first operation stored in said entry of said trace cache is not in said execution path according to said prediction, assign to said first operation a non-executable status indicative that said first operation is not in said execution path;

detect that said prediction is incorrect subsequent to assigning said non-executable status to said first operation; and

assign an executable status to said first operation in response to said detecting, wherein said executable status is indicative that said first operation is in said execution path.

10. The microprocessor as recited in claim 9, further comprising a scheduler coupled to receive said first operation from said dispatch logic and configured to store an indication of said non-executable status of said first operation.

11. The microprocessor as recited in claim 10, wherein said scheduler is further configured to prevent said first operation from executing in response to storing said indication of said non-executable status of said first operation.

12. The microprocessor as recited in claim 10, wherein said scheduler is further configured to issue said first operation for execution without said dispatch logic refetching said first operation from said trace cache in response to said dispatch logic assigning said executable status to said first operation.

13. The microprocessor as recited in claim 9, wherein said dispatch logic is further configured to:

determine a destination of said first operation in response to assigning said executable status to said first operation;

determine that a second operation is dependent upon the destination of said first operation in response to determining said destination; and

configure said second operation to receive a result from said first operation in response to determining that said second operation is dependent upon the destination of said first operation.

14. The microprocessor as recited in claim 13, wherein said dispatch logic is further configured to store a respective destination specified by each unretired operation in a destination list and to determine said destination of said first operation by accessing said destination stored in said destination list.

15. The microprocessor as recited in claim 9, wherein said dispatch logic is further configured to:

in response to predicting said execution path, if said first operation stored in said entry of said trace cache is in said execution path according to said prediction, assign said executable status to said first operation;

detect that said prediction is incorrect subsequent to assigning said executable status to said first operation; and

assign said non-executable status to said first operation in response to said detecting.

16. The microprocessor as recited in claim 15, wherein said dispatch logic is further configured to:

determine a destination of said first operation in response to assigning said non-executable status to said first operation responsive to detecting that said prediction is incorrect;

determine that a second operation is dependent upon the destination of said first operation in response to determining said destination; and

configure said second operation to receive a result from a source other than said first operation in response to determining that said second operation is dependent upon the destination of said first operation.

17. The microprocessor as recited in claim 9, wherein said branch prediction logic is further configured to predict an execution path of a second conditional branch operation stored in said entry of said trace cache, wherein said dispatch logic is further configured to determine that said first operation is dependent upon said first conditional branch operation and said second conditional branch operation, and wherein said dispatch logic is further configured to assign an executable status to said first operation in response to detecting that said prediction of said first conditional branch is incorrect dependent upon said first operation being in the predicted execution path of said second conditional branch operation.

18. A computer system, comprising:

a system memory; and

a microprocessor coupled to the system memory, wherein the microprocessor comprises:

a trace cache comprising a plurality of entries, wherein each entry is configured to store one or more operations;

branch prediction logic configured to predict an execution path of a first conditional branch operation stored in an entry of a trace cache; and

dispatch logic coupled to said branch prediction logic and to said trace cache and configured to:

if a first operation stored in said entry of said trace cache is not in said execution path according to said prediction, assign to said first operation a non-executable status indicative that said first operation is not in said execution path;

detect that said prediction is incorrect subsequent to assigning said non-executable status to said first operation; and

assign an executable status to said first operation in response to said detecting, wherein said executable status is indicative that said first operation is in said execution path.

19. The computer system as recited in claim 18, wherein said microprocessor further comprises a scheduler coupled to receive said first operation from said dispatch logic and configured to store an indication of said non-executable status of said first operation.

20. The computer system as recited in claim 19, wherein said scheduler is further configured to prevent said first operation from executing in response to storing said indication of said non-executable status of said first operation.

21. The computer system as recited in claim 19, wherein said scheduler is further configured to issue said first operation for execution without said dispatch logic refetching said first operation from said trace cache in response to said dispatch logic assigning said executable status to said first operation.

22. The computer system as recited in claim 18, wherein said dispatch logic is further configured to:

determine a destination of said first operation in response to assigning said executable status to said first operation;

determine that a second operation is dependent upon the destination of said first operation in response to determining said destination; and

configure said second operation to receive a result from said first operation in response to determining that said second operation is dependent upon the destination of said first operation.

23. The computer system as recited in claim 22, wherein said dispatch logic is further configured to store a respective destination specified by each unretired operations in a destination list and to determine said destination of said first operation by accessing said destination stored in said destination list.

24. The computer system as recited in claim 18, wherein said dispatch logic is further configured to:

in response to predicting said execution path, if said first operation stored in said entry of said trace cache is in said execution path according to said prediction, assign said executable status to said first operation;

detect that said prediction is incorrect subsequent to assigning said executable status to said first operation; and

assign said non-executable status to said first operation in response to said detecting.

25. The computer system as recited in claim 24, wherein said dispatch logic is further configured to:

determine a destination of said first operation in response to assigning said non-executable status to said first operation responsive to detecting that said prediction is incorrect;

determine that a second operation is dependent upon the destination of said first operation in response to determining said destination; and

configure said second operation to receive a result from a source other than said first operation in response to determining that said second operation is dependent upon the destination of said first operation.

26. The computer system as recited in claim 18, wherein said branch prediction logic is further configured to predict an execution path of a second conditional branch operation stored in said entry of said trace cache, wherein said dispatch logic is further configured to determine that said first operation is dependent upon said first conditional branch operation and said second conditional branch operation, and wherein said dispatch logic is further configured to assign an executable status to said first operation in response to detecting that said prediction of said first conditional branch is incorrect dependent upon said first operation being in the predicted execution path of said second conditional branch operation.